state to I/O state due to removal of its interval from cache, a selected Resource Model Equation (*e.g.*, Resource Model Equation (18)) may be used in step 320 to calculate a lower-bound value of cycle time T (*i.e.*, value calculated from I/O capacity function or left-hand side of the equation) and an upper-bound value of cycle time T (*i.e.*, value calculated from buffer space function or right-hand side of the equation), with both calculations made including the new or returning viewer

$$\text{Lower Bound} = MaxNoV\_perDisk * AA / [1 - Reserved\_Factor - MaxAggRate\_perDisk / TR]$$

$$\text{Upper Bound} = (1 - Reserved\_Factor) * B_{max} / \{Buffer\_Multiplcity * [(1 - B\_Save) * (\Sigma_{i=1}^{Nov} P_i)]\}$$

Lower and upper bound values of cycle time T calculated in step 320 may be compared in step 330 to determine whether or not the lower-bound value is less than the upper-bound value (*i.e.*, to determine whether or not the values of I/O capacity and buffer space overlap or whether or not a possible value or range of possible values of cycle time T exist to balance said I/O capacity with said buffer memory space) or, for example, whether or not the lower bound is less than the upper-bound value by a pre-determined threshold amount or range of values (*e.g.*, Upper Bound – Lower Bound ≥ 0.05). In either case, if the specified relationship is not true (*i.e.*, lower bound value is greater than or equal to upper-bound value, or Upper Bound – Lower Bound is not ≥ 0.05), then storage management processing engine 105 may refuse to admit the new or returning viewer to its I/O task pool and the values of cycle time T and read-ahead size N_i may be left unchanged as shown in step 340. However, if in step 330 the calculated lower bound value is less than the calculated upper-bound value (*i.e.*, values of I/O capacity and buffer space do not overlap), then storage management processing engine 105 may admit the new viewer or returning viewer to its I/O task pool as shown in steps 360 and 380, after determining in step 350 whether the existing cycle time T and read-ahead size N_i need to be modified to take into account the new or returning viewer.

In step 350, the existing value of cycle time T may be compared to the range between the newly determined lower bound and the newly determined upper bound calculated in step 320. In

step 360, the new or returning viewer may be admitted, and the existing value of cycle time T and the existing read-ahead size may be left unchanged, if the existing value of cycle time T falls within the range between the newly determined lower bound and the newly determined upper bound. However, if the existing value of cycle time T falls outside the value range between the newly determined lower bound and the newly determined upper bound, then a new value of cycle time T may be selected or otherwise determined in step 370 from the range of values existing between the newly determined lower and the upper bounds. This newly determined cycle time T may then be used in step 370 (*e.g.*, by storage management processing engine 105) to determine a new value of read-ahead segment size N_i for all existing viewers using, for example, equation (12) and the new or returning viewer admitted in step 380.

Alternatively, in those embodiments employing a Resource Model Equation that uses a value of Reserved_Factor, then an attempt may be made to modify the existing value of Reserved_Factor to allow admittance of an existing viewer that is returning from cache so as to minimize disruption to the returning viewer if the specified relationship found not true in step 330, as shown in the exemplary embodiment of FIG. 3B. As shown in FIG. 3B, if it is determined in step 332 that a new viewer is being introduced, then storage management processing engine 105 may refuse to admit the new viewer to its I/O task pool and the values of cycle time T and read-ahead size N_i may be left unchanged as shown in step 340. However, if the viewer is determined in step 332 to be an existing viewer returning from cached state to I/O state, then value of Reserved_Factor may be reduced by a given amount in step 334 (*e.g.*, from a value of about 0.2 to a value of about 0.1), and then lower and upper bound values of cycle time T re-calculated in step 336 using the reduced value of Reserved_Factor. The recalculated lower and upper bound values of cycle time T calculated in step 336 may be compared in step 338 to determine whether or not the lower-bound value is less than the upper-bound value, or whether or not the lower bound is less than the upper-bound value by a pre-determined threshold amount. In either case, if the specified relationship is not true, then storage management processing engine 105 may refuse to admit the returning viewer to its I/O task pool and the values of cycle time T and read-ahead size N_i may be left unchanged in step 340. However, if the specified relationship is found true in step 338, then storage management processing engine 105 may

admit the returning viewer to its I/O task pool in a manner similar to that as described in relation to FIG. 3A, *i.e.,* as shown in steps 360 or 380, after determining in step 350 whether the existing cycle time T and read-ahead size N_i need to be modified in step 370 to take into account the returning viewer. While the value of Reserved_Factor may be reduced in step 334 in an attempt to admit a viewer/s returning from cached state, in one exemplary embodiment the original value of Reserved_Factor is not changed and is still used for admission control when accepting a new viewer/s.

In the practice of the disclosed methods and systems, cycle time T may be any value selected from between the determined lower and upper bounds of a selected Resource Model Equation, such as Resource Model Equation (18). However, in those situations when an information management system is lightly loaded, there may be a big gap between the upper bound and the lower bound, meaning that there is a large range from which a particular value of cycle time T may be selected. If the value of T is selected to be too high within the range, then it is possible that exhaustion of the buffer space may occur while there is still excess I/O capacity remaining. Conversely, if the value of T is selected to be too small within the range, then it is possible that exhaustion of I/O capacity may occur while there is still excess buffer space remaining. In either case, an unbalanced resource utilization may result that requires re-determination of the cycle time T value, and a resulting modification of the read-ahead size based on the new cycle time T. To address this concern, one exemplary embodiment may optimize system performance by using an admission control policy that selects a value of cycle time T in a manner that helps ensure substantially balanced utilization of pre-allocated I/O resource and buffer space by, for example, increasing or maximizing the elapsed time during which introduction of new viewers will not force the redefinition of cycle time T. This may be done in any suitable manner, for example, based on empirically derived information. To illustrate, if a maximal cycle time T value is empirically determined to be about 25 seconds, run time selection of cycle time T value may be determined based on the following relationship:

$$T = \min(\text{lower\_bound} + 20, (\text{lower\_bound} + \text{upper\_bound})/2) \qquad (20)$$